REMARKS

Claims 116-163 are pending.

Claims 116-163 stand rejected.

Claims 116, 128, 137, 138, 140, 146, 147, 149, 153, 155, 156, and 158 have been amended. No new matter has been added. Support for these amendments can be found, at least, within the originally-filed specification on page 2, lines 1-11, and page 18, line 5 through page 19, line 20.

Rejection of Claims under 35 U.S.C. § 103

Claims 116-162 stand rejected under 35 U.S.C. § 103(a) as purportedly being unpatentable over U.S. Patent No. 6,438,542 ("Koo") in view of U.S. Patent No. 6,523,028 ("DiDomizio"). Applicants respectfully traverse this rejection.

Applicants respectfully submit that Koo and DiDomizio, alone or in any combination, fail to teach or suggest, all the elements of claim 116, including (1) automatically generating a set of SQL statements that are based, at least in part, upon a received SQL statement; (2) producing a first and second result set by querying a first and second table from automatically generated SQL statements; and (3) joining the first and second result sets to produce a third result. Independent claims 128, 137, 146, 155, and 156 recite comparable limitations.

Koo Fails to Teach Generating SQL Statements Based Upon a Received SQL Statement

The Office Action characterizes Koo as presenting a method for optimizing a database query by analyzing the query to identify any joins within the query that are lossless and any tables of the identified joins that are eligible for removal. Koo then rewrites the query to eliminate the identified tables that are eligible for removal. Among other deficiencies of Koo, Applicants submit that given database tables, Koo's query optimization does not generate SQL statements that will be performed on the tables, as claimed. Koo's method is simply as Koo describes it, optimizing a query by identifying tables that can be removed prior to performing the query, and then rewriting the query into a simpler form. In fact, Koo is not directed toward execution of any queries, merely analysis and optimization prior to execution.

Applicants respectfully submit that, among other infirmities, the connection between a received SQL statement and the claimed generating of a set of SQL statements to eliminates any potential characterization of Koo's analysis as somehow teaching, showing, or suggesting the claimed generating. This lack of teaching in Koo comes as no surprise, as Koo's analysis is not intended to generate SQL statements. On the contrary, Koo is interested in eliminating portions of a given query in order to produce an optimized query.

Koo's method begins with an original query and finishes by producing a rewritten query. Producing the rewritten query is Koo's end result; Koo executes neither the original query, nor the rewritten query. By contrast, the claimed method automatically generates a set of SQL statements, and produces two result sets by then querying the two tables.

Thus, Applicants submit that Koo's method is complete before any queries are executed, whereas the claimed method is integrated with the execution of queries. In other words, the domains in which Koo's method and the claimed method are relevant do not overlap. Thus, Koo's query rewriting avoids ever performing any query on tables removed by the optimization process.

Koo Fails to Teach Producing Result Sets by Querying a First and Second Table

Applicants respectfully submit that Koo does not produce result sets from querying the tables using generated SQL statements, where the result sets are subsequently used in the performance of a join operation, as claimed. An example of Koo's method performs a rewrite of the following join query:

```
SELECT COUNT(*)
FROM STARS ACCOUNT A, STARS.CUSTOMER C
WHERE A.CUSTID = C.CUSTID AND A.BALANCE < 10
```

Koo 5:5-10. Koo's example join query operates on two tables, a CUSTOMER table, and an ACCOUNT table. Given the above join query, Koo performs an analysis to determine that the CUSTOMER table can be eliminated given the join conditions. *See*, Koo 5:11-22. After eliminating the CUSTOMER table, Koo rewrites the above query as the following:

```
SELECT COUNT(*)
FROM STARS ACCOUNT A
WHERE A.BALANCE < 10
```

Koo 5:22-25. In other words, Koo's system rewrites the query, without the one or more tables, and <u>without ever performing any kind of SQL query on either table</u>.

As noted above, Koo's rewriting process does not perform any queries on the CUSTOMER or ACCOUNT tables. Thus, it must be the case that Koo fails to perform queries using any generated SQL statements. At least one reason Koo fails to teach or suggest this limitation is that the CUSTOMER table in Koo is <u>eliminated</u> before either the original query or the rewritten query is performed. As described above, Koo is not performing any queries on the tables, but is instead identifying which tables can be eliminated in order to rewrite the query in a simpler form. Koo's determination of which tables to eliminate does not involve generating a set of SQL statements to query a first and second table, as claimed. Even if Koo could otherwise be successfully analogized to the claimed method, at the moment Koo eliminated any table, the analogy would fail because the claimed method performs queries on every table received.

<u>Koo Fails to Teach Joining First and Second Result Sets to Produce a Third Result Set</u>

Applicants respectfully submit that no part of the cited sections of Koo's method can be interpreted as producing result sets from performing any manner of SQL statements on any tables, particularly where such result sets are then used in the performance of a join operation, as claimed. As noted above, because Koo does not perform any queries on the tables that are objects of the original query, it necessarily follows that <u>Koo does not perform any query on the tables using generated SQL statements</u>.

In addition to citing the earlier-presented example in Koo, the Office Action cites to a section of Koo, characterized by the Office Action as presenting different kinds of joins that can be analyzed and rewritten, as purported disclosure of the claimed querying a table to produce a result set, where the query uses a set of generated SQL statements. *See*, Office Action, pp. 3 and 4 (citing Koo 7:10-35). Applicants submit that this section merely details which types of joins can be analyzed using Koo's method. This section of Koo is not an alternative of the above-cited example. Instead, this section presents the details of how the underlying principles of Koo's method, described earlier, operate. Thus, a mere listing of types of joins that can be analyzed by Koo's method in no way teaches or suggests any additional limitations, nor provides any further meaningful disclosure, and so, also fails to show, teach or suggest the claimed production of a result set by querying a table with a set of generated SQL statements.

In addition to the above sections of Koo, the Office Action cites to an example within the "Column Equivalents Predicates" section as purported disclosure of the claimed joining of result sets (result sets produced by performing generated SQL statements on first and second tables). *See*, Office Action, p. 4 (citing Koo 6:40-50). Applicants submit that this section also merely provides the manner in which the earlier-described analysis is performed (these cited sections are within a "Fundamentals" section). This section of Koo does not present an alternative to the earlier-cited example. Instead, this section simply presents aspects of the logical predicates that are used in certain steps of Koo's analysis of the original query, as described earlier. Thus, this section also fails to provide any meaningful additional disclosure beyond, or any alternatives to, the previously cited section, and so is incapable of showing, teaching, or suggesting any method or aspect different from that already disclosed in Koo. Thus, Koo fails to teach or suggest the claimed joining of first and second result sets in producing a third result set, and in fact, is incapable of so doing.

Thus, in an attempt to demonstrate the teaching or suggestion of the above-noted (and distinctly recited) limitations, the Office Action cites to two sections of Koo that disclose no more than the sections of Koo already cited against other of the claim limitations. While the citation of different passages of a reference, in rejecting a claim's limitations, is clearly possible (and common), sections cited against distinct limitations must similarly provide teachings that are also distinguishable from one another, by definition. In the present scenario, the first set of cited sections of Koo present an example of rewriting an original query, while the second set of cited sections explain the principles by which the original query is analyzed and rewritten. In other words, whatever teachings these sections provide, such teachings are effectively indistinguishable from one another, for purposes of teaching two or more distinct limitations. Logically, then, because the same teaching cannot be used as purported disclosure of two or more distinct limitations, Koo must fail to disclose all the limitations of the claims.

DiDomizio Fails to Teach the Claimed Generating, Producing, and Joining

The Office Action does not cite DiDomizio as teaching the claimed producing and joining limitations. Applicants agree with the Examiner that DiDomizio fails to apply to the claimed producing and joining limitations, and further, that DiDomizio also fails to teach or suggest the claimed automatically generating a set of SQL statements that are based, at least in part, upon a received SQL statement.

DiDomizio is directed to a system that allows a user to access multiple databases, where each database may have a different structure. *See* DiDomizio, Abstract. In DiDomizio's system, a user is presented with an interface for the user to enter unstructured, English-language keyword search. *See* DiDomizio, 4:7-29. DiDomizio's system then generalizes the unstructured search to query a plurality of databases with potentially matching tables. *See* DiDomizio, 4:30-62. This initial step in DiDomizio presents the first distinction (of several) between the claimed invention and DiDomizio – DiDomizio receives an unstructured search, whereas the claimed invention receives a SQL statement.

DiDomizio's system then presents the user with results from the initial database queries, and the user selects the relevant components from which the final, results-producing database query will be created. A second distinction between DiDomizio and the claimed automatic generating of a set of SQL statements is that the claimed method is performed without user intervention. By contrast, DiDomizio's creation of a database query is accomplished by user-directed selections of the components from which to create the database query. DiDomizio's manual process is contrary to the automatic generation of SQL statements.

Further, not only does the manual aspect of DiDomizio fail to teach the claimed automatic generating of SQL statements, DiDomizio's system also fails to teach or suggest automatically generating a set of SQL statements by using relationships between the first and tables, as claimed. The basis for the creation of DiDomizio's database query is simply the user selections from which the database query is to be created. Thus, DiDomizio fails to teach or show the claimed automatic generation of a set of SQL statements by using a relationship to perform the automatic generation of the set of SQL statements.

For at least these reasons, Applicants submit that Koo fails to teach or suggest all the elements of independent claims 116, 128, 137, 146, 155, and 156, and all dependent claims. Applicants therefore respectfully request the Examiner's reconsideration and withdrawal of the rejections to these claims.

CONCLUSION

In view of the amendments and remarks, the application and the claims therein are believed to be in condition for allowance. Nonetheless, should any issues remain that might be subject to resolution through a telephonic interview, the Examiner is invited to telephone the undersigned.

If any extensions of time under 37 C.F.R. § 1.136(a) are required in order for this submission to be considered timely, Applicants hereby petition for such extensions. Applicants also hereby authorize that any fees due for such extensions or any other fee associated with this submission, as specified in 37 C.F.R. § 1.16 or § 1.17, be charged to Deposit Account 502306.

Respectfully submitted,

/ Samuel G. Campbell III /

Samuel G. Campbell III
Attorney for Applicants
Reg. No. 42,381
Telephone: (512) 439-5084
Facsimile: (512) 439-5099